仮想化技術を用いたセキュアクライアントの提案

山本　一樹　　安井　浩之　　横山　孝典
東京都市大学

1. はじめに

近年，セキュリティ対策や総所有コストの削減のために，シンクライアントを導入する事例が増えている．しかし，シンクライアントの導入には専用の端末や高性能なサーバを必要とすることが多く，資金的制約の大きい中小企業や教育機関などでは導入が困難である．そこで我々は，仮想化技術を用いて，導入済みのクライアントPC上に，仮想的なシンクライアント端末を作成し，ネットワークブート方式のシンクライアントを実現するシステムを提案している[1].

本報告では，セキュリティ対策のための認証機能の提案と，本システムを導入した場合の性能評価について述べる．

2. 提案システムの概要

本システムでは，図1に示すように、仮想化技術を用いてクライアントPC上に仮想マシンを作成し，異なる特徴をもつ環境を動作させる．

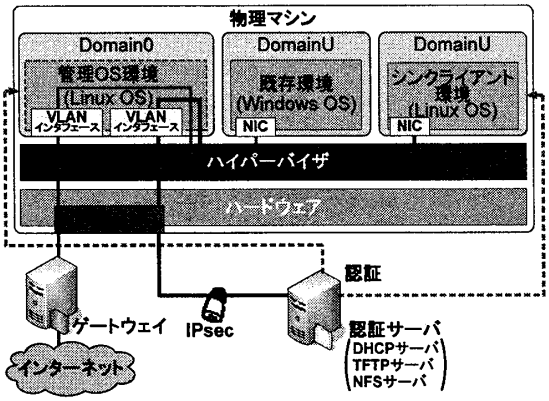

図1 システム構成

シンクライアント環境が動作する仮想マシンは，クライアントPCに物理的に搭載されている記憶装置や周辺機器を利用できないように制御しており，情報をローカルディスクやUSBメモリに保存することはできない．この仮想マシンを用いて，NFSサーバ上からOSやアプリケーションを起動することで，ネットワークブート方式の仮想シンクライアント端末を実現する．

一方，既存環境が動作する仮想マシンは，クライアントPCに物理的に搭載されているハードウェアをすべて利用可能としているので，ローカルディスク上に個別の環境を構築可能である．従来のシンクライアントではシステムを導入すると，個別の環境を構築するのは困難であったが，本システムでは，シンクライアント環境の情報を保護しつつ，1台のクライアントPCでユーザ独自の環境も構築可能である．

管理OS環境は，クライアントPCのハードウェアや仮想マシンを制御するためのOS（管理OS）が動作しているため，管理OS環境自体がセキュアでないとクライアント全体のセキュリティが保証できなくなる．そこで，外部からの脅威に対しては，後述するシンクライアント環境の認証に必要な通信以外のパケットをすべて破棄することで，侵入経路自体を塞ぐ．また，ユーザ自身が脅威になるケースも考えられるため，ユーザには他の仮想マシンを操作するための専用アプリケーションのみを与え，管理OSに命令を送るためのコンソールなどを与えないことで，ユーザによる管理OS環境の改変を防ぐ．

3. シンクライアント環境の認証

本システムでは，記憶装置や周辺機器を持たない仮想的なシンクライアント端末を用いることで，シンクライアント環境上の情報が漏洩しないことを保証している．また，管理OS環境自体に手が加えられないようにすることで，クライアントPC上で不正な仮想マシンが動作することを防ぐ．しかし，仮想化ソフトウェアや管理OS自体を入れ替えられると，それらを保証することができなくなる．

そこで，ネットワークブートを行う前に，管理OS環境と仮想シンクライアント端末の状態をサーバ側で認証することで，正常な管理OSから作成された，正常な仮想シンクライアント端末にのみ，ネットワークブートを許可する仕組みを提案する．

4. 認証機能の実装方法

認証機能の実装には，Linuxの標準的なファイアウォール機能であるiptablesとDHCPサーバの設定を動的に変更可能なOMAPI[2]を用い，図2に示す手順で認証を行う．

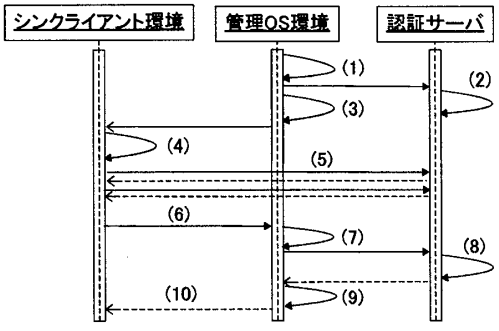

図2 認証シーケンス

まず，シンクライアント環境が起動する前に，(1)仮想シンクライアント端末のMACアドレス(***MAC***)，IPアドレス(***IP***)，NFSサーバへの通信を待ち受けるポート番号(***NFSport***)を管

A Proposal of Secure Client System using Virtualization Technology
Kazuki Yamamoto, Hiroyuki Yasui, Takanori Yokoyama
Tokyo City University

理 OS 環境で毎回異なる値になるよう動的に決定し，その内容と認証情報を認証サーバに送信する．この認証情報には，管理 OS 環境で動作するプロセス，ネットワーク構成，仮想マシンの作成に用いる設定ファイルの内容を用いる．そして，認証情報を受け取ったサーバは，(2)正当性の検証を行い，OMAPI によって，***MAC*** を持つクライアントに対し，***IP*** を割り当てるように DHCP サーバの設定を変更する．

認証サーバ上の NFS サーバは通常の 2049 番ポートで動作させるが，iptables であらかじめこのポートを閉じておく．そして，認証を受けた後，iptables に，***IP*** から送信された ***NFSport*** 宛のパケットを通常の 2049 番ポート宛に転送させるルールを追加する．したがって，認証を行っていないクライアントから NFS サーバにマウント要求を送ることができなくなる．

また，管理 OS 環境では，(3)iptables に，***IP*** から送信される 2049 番ポート宛のパケットを ***NFSport*** 宛に書き換えるルールを追加する．これによりシンクライアント環境や NFS サーバの設定に手を加えることなく通信が可能になり，認証を実現する．

クライアント側とサーバ側で iptables の設定が終わった後は，(4)***MAC*** を持つ仮想シンクライアント端末を作成し，起動させる．起動した仮想シンクライアント端末は，(5)DHCP サーバから ***IP*** を受け取って，通常のネットワークブートシーケンスでブートし，(6)NFS サーバにマウント要求を出す．この要求パケットは，(7)～(9)管理 OS 環境と認証サーバの iptables で適宜書き換えられ，(10)マウント応答を受け取ることができる．

なお，現在，認証機能は実装作業中である．

5. 実験と評価

本システムを導入した場合の性能評価のため，表 1 に示した機器を用いて，1Gbps のネットワーク上で，以下の実験を行った．

- 実際の NIC と仮想化された NIC を用いて NFS サーバとの通信速度を測定
- シンクライアント環境が起動する（ネットワークブートエージェントが起動してからログイン画面が表示される）までの時間を測定
- 円周率 419 万桁の処理時間を測定

これらの実験結果を表 2～4 に示す．なお，仮想化ソフトウェアには Xen3.4.2 を，管理 OS には Debian5.0 を用いた．

表 1 使用機器と仮想マシンへのリソース割り当て

	CPU	メモリ	NIC
サーバ	Athlon 64 X2 2.20GHz×2	2048MB	1Gbps
クライアント	Core2Duo 3.16GHz×2	2048MB	1Gbps
仮想マシン（シンクライアント環境）	Core2Duo 3.16GHz×1	512MB	100Mbps

*OS はすべて CentOS5.4

表 2 NFS サーバとの平均通信速度

	実 NIC		仮想 NIC	
	上り	下り	上り	下り
通信速度	778Mbps	940Mbps	147Mbps	556Mbps

表 3 平均起動時間

	*A	*B	*C
起動時間	58.8 sec	52.0 sec	55.6 sec

表 4 円周率の平均処理時間

	*A	*C
処理時間	68.0 sec	71.6 sec

*A ローカルディスクから直接起動

*B ネットワークブート（実ハードウェア）

*C ネットワークブート（仮想ハードウェア）

表 2 から，仮想 NIC は実 NIC に比べると，大幅に通信速度が低下していることがわかる．これは，表 1 に示したように，仮想 NIC は，実際の NIC より低速なものをエミュレートしているためである．しかし表 3 を見ると，実ハードウェアと仮想ハードウェアのネットワークブートにかかる時間の差はほとんどないことから，仮想 NIC でも十分な性能であると考えられる．ただし，この起動時間は，仮想マシンを作成するための，管理 OS の起動時間（30 秒程度）を考慮していないため，ユーザが実際に操作可能になるまでには，実ハードウェアに比べて約 1.5 倍の時間を必要とする．

また，表 4 から，仮想ハードウェアでの処理では 5%程度の遅延があることがわかる．しかし，仮想化によるオーバーヘッドはそれほど大きくなく，提案するシステムでシンクライアントを実現しても，作業効率が著しく低下することはないと考えられる．

6. まとめ

本報告では，シンクライアント環境や NFS サーバの設定に手を加えることなく，シンクライアント環境を認証する手法を提案した．この手法を用いることで，従来のネットワークブート環境の構築・運用方法を，そのまま仮想環境に移行することが可能になる．また，システムの性能評価から，仮想シンクライアント端末を利用した場合でも，実運用可能な性能を期待できることがわかった．今後は，認証機能の実装完了と，管理 OS 環境の構成調整による，起動時間短縮を目指す．

参考文献

[1] 山本一樹，安井浩之，横山孝典："仮想化技術を用いたセキュアクライアントの提案"，コンピュータセキュリティシンポジウム 2009 論文集［第一分冊］，pp.135-140，2009

[2] omapi(Linux Reviews)
http://linuxreviews.org/man/omapi/